SYSTEM AND METHOD FOR SEARCHABLE INSTANT MESSAGING CHAT REPOSITORIES USING TOPIC AND IDENTIFIER METADATA

CROSS-REFERENCE TO RELATED APPLICATION

[0001] The present invention is related to the subject matter of U.S. Patent Application _____ (Attorney Docket number _____), incorporated herein by reference.

FIELD OF THE INVENTION

[0002] The present invention relates generally to Instant Messaging, and specifically to storing structured "chats" in a filtered, searchable repository on either a client computer, a server computer or both the client computer and the server computer.

BACKGROUND OF THE INVENTION

[0003] Instant Messaging (IM) is a communication service that allows a user on one computer to send and receive text messages in real time with another user on a second computer connected to the first computer by the Internet or other distributed communications protocol. In one version of instant messaging, the channel for this communication is often referred to metaphorically as a chat room, and the communication over the channel is referred to as chat. In another version, two or more users engage in an instant messaging conversation outside the boundaries of a chat room; each user is able to view a chat transcript on one or more local machines. Each user has a list of other users with whom they may want to chat. The IM service may alert the user when another user, who is on the user's participant list, is also online. A chat

may then be started, and once started, either user can enter text that will appear on other user's displays.

[0004] IM is offered by most network and online service providers. In the workplace, IM conversations often take place between people working on common projects. In engineering and computer design activities it is not uncommon for one user to have ten or more multiple chats running simultaneously. Chats may describe simple information shared by two workers scheduling a meeting, or chats may contain complex discussions regarding proprietary information and critical decisions to keep a project moving forward. Chats may contain a high degree of historical data and proprietary knowledge that is useful not only to the participants but that may be useful to other workers within the organization who did not participate in the chat. Moreover, the knowledge and data captured in the chat transcript may be useful as historical data for future reference by the participants as well as others. Finally, in some areas, such as financial services, preservation of the chat transcript may be required by securities regulations. Therefore, it is important for organizations to preserve information captured in chat transcripts so that both the participants and the organizations can use the knowledge, information, and decisions contained in these IM conversations.

[0005] The prior art discloses both monitoring of chats as well as common repositories for chat transcripts. Monitoring of chats is disclosed in two United States patents. United States Patent 6,631,412 to Glasser et al. discloses a system for monitoring user activity in an instant messaging session if the user has actively entered data during a first time interval. A message processor forms one or more data packets for transmission on the network such as user-entered data, and user activity messages. A user activity monitor measures the time period since the last user activity. United States Patent 6,076,100 to Cottrille et al. discloses a method for monitoring

chat room conversations including a database for storing penalty information corresponding to the identity of chat users.

[0006] The commercial prior art discloses methods to save IM conversations on both the server side and the client side. On the server side, users can save their IM messages in a common repository. Examples of applications having common repositories are FACETIME, IMLOGIC, MESSAGERITE, and IRON MOUNTAIN. These IM services provide archiving services so that financial services firms can demonstrate compliance with securities regulations. These solutions log only a limited number of end users in high-risk job roles. The financial services archiving, for example, applies only to particular groups in specific job roles. On the client side, ICQ provides a client-side repository. ICQ provides the ability to play back a chat session, both for the user and for new chat participants. But the ICQ repository is local, not distributed. The ICQ repository also depends on the affirmative choice of the person who saved the chat to play it back; therefore, in order to share the saved chat, the person must either be online, or must move the contents of the chat into a non-IM context such as email, web page, or text document. Another person can not actively open or search the chat unless the first person plays it back. Other applications, such as LOTUS SAMETIME, allow saving IM conversations as flat text. Both kinds of client applications place responsibility on the end users to save chat.

[0007] While the repositories described above allow full text search of the selectively logged chat, a number of needs arise that are not met. Server side repositories have at least three limitations. First the prior art repositories do not provide topic-based filtering by end users. Second, while chat transcripts may indeed be logged for the purpose of security and auditing, these repositories do not provide a commonly accessible, searchable repository. In other words, the prior art repositories are limited to full text online searches essentially by security personnel

or senior management. Therefore, the knowledge generated or repeated in IM conversations isn't directly searchable or shareable across organizations. Chat clients aren't able to reference other chats and to recall them from a repository. Not all conversations are worth saving, but organizations should at least have the ability to choose whether to save all or some chats, or all or some chats on particular topics. Third, the prior art repositories do not allow filtering or searching on marked topics, nor do they log relevant metadata.

[0008] Client side repositories have at least three limitations. First, the end users are responsible for operations with the chat transcript, such as operations for cutting and pasting into email, for saving their chat transcripts, or for posting the transcripts to message boards. Leaving such actions up to each individual end user is an inconsistent process not subject to automation or to the application of business rules. Moreover, allowing users to control the save function, as well as to cut and paste the transcript, is an unreliable process that can also pose security risks due to unsupervised editing of the transcript. Second, personal saving of chats means the saved chats are not accessible among larger groups that might be interested in the content of the chat. Third, all chat participants do not have equal access to the chat history if the participants do not save the chat. Participants unfamiliar with previous conversations are often called into a new chat. Most chat applications do not provide search or filtering of a repository, or direct access to transcripts. Therefore, new participants must be given background and orientation to the topic at hand by second hand accounts (rather than the initial chat transcript). These additional messages take up time and resources.

[0009] The chat applications that do provide search or filtering of a repository have limitations. One method uses string-matching to provide search results. Yet the same string can occur within the context of many different topics. Second, known search methods do not filter

based on a person's membership in a group, such as an organization, and so the abstract characteristics of who is speaking are not available as a filter. For example, a manager may want to see what people in his or her organization are chatting about, or the manager may want to search for a particular phrase within chats on a defined topic. Currently known methods do not allow for search and filtering of cross-reference among chats, even though many chat transcripts refer to previous conversations.

[0010] Therefore, a need arises for a system and method to store structured "chats" in a searchable repository on either or both of the client computer and the server computer. A need further arises for more precise searches through filtering by metadata, such as chat topic and user characteristics or by Lightweight Directory Access Protocol (LDAP) data such as organization, project, or membership. A need also exists for a system and method to auto-alert and/or to export to other formats such as e-mail or content management based upon the chat topics and/or participants. Moreover, a need arises to allow for retrospective application of topics to chat transcripts that have not been tagged for particular topics by end users. In this fashion, an organization may recover information in chat transcripts according to its own needs regardless of how end users have classified that information. Organizations may also automate the tagging of chats by topic if needed. Finally, a need exists for a system and method to allow managed decisions for security purposes to specify what chats will be saved and to provide for an automatic saving of specified chats or chats on specified topics.

SUMMARY OF THE INVENTION

[0011] The present invention, which meets the needs stated above, is a system and method for storing IM text in a repository where it can be searched by topic or participant based filters. The repository may be a centralized database on the server side or it may be distributed in

the client environment. The system comprises an IM Topic (IMT) program, an IM Filter (IMF) program, an IM save program (IMS), and an IM automatic alert (IMAL) program. The IMT program assigns topic tags to turns and topic shifts in the IM conversation. The IMF program searches the repository based on user selection of filters as well as full text search. There may be any number of topic tags. The IMS program automatically scans IM transcripts for topic tags and identifiers and saves transcripts or transcript segments that match pre-selected topic tags or identifiers to the repository. The IMAL program sends notifications and/or exports transcripts or transcript segments to pre-designated addresses when a match is made between a topic or identifier in a transcript and a pre-selected topic or identifier.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] The novel features believed characteristic of the invention are set forth in the appended claims. The invention itself, however, as well as a preferred mode of use, further objectives and advantages thereof, will best be understood by reference to the following detailed description of an illustrative embodiment when read in conjunction with the accompanying drawings, wherein:

[0013] FIG. 1 is an illustration of a computer network used to implement the present invention;

[0014] FIG. 2A is an illustration of a repository memory associated with the present invention;

[0015] FIG. 2B is an illustration of a client memory associated with the present invention;

[0016] FIG. 3A is an illustration of a server computer associated with the present invention;

[0017] FIG. 3B is an illustration of a client computer associated with the present invention.

[0018] FIG. 4 is an illustration of the logic of the instant message tagging program associated with the present invention;

[0019] FIG. 5 is an illustration of an instant message display;

[0020] FIG. 6A is an illustration of an instant message display with two topic segments in vertical alignment with the instant message transcript;

[0021] FIG. 6B is an illustration of an instant message display with two topic segments in vertical alignment without the instant message transcript;

[0022] FIG. 6C is an illustration of an instant message display with a topic list link and two subtopic display headings;

[0023] FIG. 6D is an illustration of an instant message display with topics, subtopics and cascading window implementation;

[0024] FIG. 6E is an illustration of an instant message display with topics and tabs for subtopic display;

[0025] FIG. 7 is an illustration of the logic of the instant messaging topic tagging program;

[0026] FIG. 8 is an illustration of a document repository with search capability;

[0028] FIG. 9 is an illustration of an instant messaging repository search program;

[0029] FIG. 10 is an illustration of an instant message repository system configuration program; and

[0030] FIG. 11 is an illustration of an instant message repository auto save program.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

[0031] As used herein, the term "chat" means communication by one person at a computer with another person at a remote computer using instant messaging (IM).

[0032] As used herein, the term "chat transcript" means the text of the data transmitted by a first participant and the text of the data transmitted by a second participant during a chat.

[0033] As used herein, the term "computer" shall mean a machine having a processor, a memory, and an operating system, capable of interaction with a user or other computer, and shall include without limitation desktop computers, notebook computers, tablet computers, personal digital assistants (PDAs), servers, handheld computers, and similar devices.

[0034] As used herein, the term "identifier" shall mean any metadata indicating user characteristics and may further include, without limitation, Lightweight Directory Access Protocol (LDAP) data such as organization, project, or membership.

[0035] As used herein, the term "instant messaging" (IM) means a communication service that allows a user on one computer to send and receive text messages in real time with another user on a second computer connected to the first computer by the Internet.

[0036] As used herein, the term "subtopic" means a term used to further subdivide a topic and chat transcript or chat transcript segment.

[0037] As used herein, the term "topic" means a term or terms used to identify the subject of a chat transcript or of a chat transcript segment.

[0038] As used herein, the term "topic marker" means an identifier attached to a chat transcript segment as metadata.

[0039] As used herein, the term "turn" means a shift in textual communication during a chat indicated by successive statements.

[0040] As used herein, the term "topic shift" means a change from one topic or subtopic to another topic or subtopic in a chat where the change is indicated by selection of new topic or subtopic marker or by change to new topic input entry field and attachment of the new topic or subtopic marker to a chat transcript or chat transcript segment.

[0041] FIG. 1 is an illustration of computer network **90** associated with the present invention. Computer network **90** comprises local computer **95** electrically coupled to network **96**. Local computer **95** is electrically coupled to remote computer **94** and remote computer **93** via network **96**. Local computer **95** is also electrically coupled to server **91** and database **92** via network **96**. Network **96** may be a simplified network connection such as a local area network (LAN) or may be a larger network such as a wide area network (WAN) or the Internet or World Wide Web. Furthermore, computer network **90** depicted in FIG. 1 is intended as a representation of a possible operating network containing the present invention and is not meant as an architectural limitation.

[0042] The internal configuration of a computer, including connection and orientation of the processor, memory, and input/output devices, is well known in the art. The present invention is a methodology that can be embodied in a computer program. Referring to FIG. 2A, 2B, 3A, and 3B, the methodology of the present invention is implemented by first memory **200**, second memory **250**, server computer **91** and local computer **94**. First memory **200** contains topic file **220**, text file **210** and IM auto alert (IMAL) program **230**. Second memory **250** contains IM Topic program (IMT) **270**, IM Filter program (IMF) **260** and IM auto save program (IMS) **280**. First memory **200** may reside in server **91**, database **92**, or may be distributed in network **90**

among one or more of server **91**, local computer **94**, or remote computers **93** and **95**. Second memory **250** may reside in local computer **94** or server **91**. Alternatively, first memory **200** may be combined with second memory **250**. Further in the alternative, first memory **200** and second memory **250** may be distributed throughout network **96**. First memory **200** is connected to network **96** by local computer processor **320**. Second memory **250** is connected to network **96** by server processor **350** of server computer **91**.

[0043] FIG. 4 shows the logic of IM Topic program (IMT) **270**. IMT **270** begins (**402**) and opens chat window in local computer **91**. The chat window displays the names of participants who are on-line and with whom a chat can be conducted. The user selects a name from the list (**414**). Selection can be accomplished by clicking on the name with user input device **360**. A prompt is displayed allowing the user to identify the topic of the chat (**418**). In an alternative embodiment for an automated IMT (not shown), following step **414** of IMT **270**, the user may be asked whether automatic topic tagging is desired. In the alternative embodiment, if the user elects automatic topic tagging, then automated IMT would proceed to step **438** of IMT **270** and the chat would begin. In the alternative embodiment, a variety of techniques familiar to persons skilled in the art would be used to select topics and/or subtopics from the chat text and to automatically insert the topic tags and subtopic tags when turns and topic shifts are identified. Returning to the embodiment of FIG. 4, IMT **270** continues and a determination is made whether the user has entered a topic (**422**). If not, a determination is made whether the user has selected from a list of available topics (**426**). If the user does not enter a topic (**422**), and if the user does not select a topic from the list (**426**), then an error message is displayed and the user is again asked to enter a topic (**422**). In an alternative embodiment (not shown), when a message is displayed at step **428** of IMT **270**, the user may be presented with an

option to apply topics and/or subtopics retrospectively. In the alternative embodiment, if the user elects to apply topics and/or subtopics retrospectively, then the user will be prompted to do so after completion of the chat (after step **444** and before step **452** of IMT **270** below). In the alternative embodiment, if the user elects to apply topics retrospectively, then the alternate embodiment will proceed to step **438** of IMT **270.** In the alternative embodiment, upon completion of the chat, the user will be prompted to identify topics and/or subtopics, and upon entry of the topics and/or subtopics by the user, the alternate embodiment will automatically insert the topics and/or subtopic tags into the chat text. Returning to the embodiment of FIG. 4, if at step **422**, the user entered a topic, or if at step **426**, the user selected a topic from a list, then a determination is made as to whether the user wants to designate one or more subtopics (**430**). If so, the user enters the subtopic or subtopics (**434**) and the user begins the chat (**438**). IMT **270** monitors the chat and determines whether a turn has occurred in the chat (**444**). If not, the chat is saved **452** and IMT **270** goes to step **456**. Chats are saved in a structured format such as XML to allow metadata for the topics and subtopics to be saved. The saved data is written to a database such as database **92** (See FIG. 1) and accessed via SQL or similar language. If a turn has occurred, a determination is made as to whether a topic shift has occurred (**448**). If a topic shift has not occurred, IMT **270** continues to monitor the chat (**438**). If a topic shift has occurred, IMT **270** goes to step **418** and repeats the process of either entering a topic or subtopic, or selecting a topic or subtopic. IMT **270** continues until a determination is made at step **444** that a turn has not occurred and the chat is saved. The determination of whether a turn has occurred can be based on a pre-designated factor such as the amount of textual data entered or a time period. After the chat is saved (**452**), a determination is made whether a new chat is to be conducted (**456**). If so, IMT **270** goes to step **414**. If not, IMT **270** stops (**460**). IMT **270** can be

implemented using VB, C++, Java or AWT/Swing. IMT **270** can be implemented as a plug-in to reduce the cost of a new IM program and/or making users download and install a new IM program.

[0044] FIG. 5 depicts a chat window without the topic tag of IMT **270**. Chat window **500** has utility section **502** from which the user can select actions such as file, edit, view, action or help. Chat identifier area **504** shows with whom the chat is being conducted. Chat transcript area **506** displays the conversation between the user and the selected chat participant.

[0045] FIG. 6A depicts a chat window employing topic tags. Next to chat transcript **506**, a first transcript segment **510** for Topic A **508** and a second transcript segment **514** for Topic B **512** are displayed. FIG. 6B shows a variation with utility area **502**, Chat identifier area **504**, first transcript segment **510**, and second transcript segment **514**. FIG. 6C shows a variation with topic tag list **520** displayed at the end of the chat transcript. When topic tag list **520** is activated such as by clicking on one of the topics tags, the selected topic tag, such as topic A **508** or topic B **512**, is displayed with the corresponding transcript segment. An additional variation is shown in FIG. 6D where topic A **508** is further divided according to subtopics A1 **522** and A2 **524**. The division into subtopics can be extended to any number of levels of subtopics. FIG. 6E shows a method of displaying topics and subtopics in one area **540** above the chat transcript. In area **540** the subtopics are displayed as tabs in order to save space. The topics and tabs may be color coded and the subtopics may be in layers to maximize the use of space.

[0046] FIG. 7 displays IMT **700** which is a variation on IMT **270** that may be implemented in conjunction with IMT **270** by the user who initiated the chat, or by a chat participant. IMT **700** starts (**702**) and a participant is selected for a chat (**704**). A topic tag for the chat is displayed (**706**). A determination is made as to whether the user wants to accept the

topic tag (**708**). If so, the chat will begin (**718**). If not, a determination is made whether the user wants to distinguish the topic tag (**710**). If so, the user will enter a term for the topic tag (**712**). A determination is made whether the other chat member accepts the topic tag entered by the user (**714**). If not, a default topic tag is used (**716**), and the chat begins (**718**). If the user does not want to distinguish the topic tag, the chat will begin (**718**). A determination is made as to whether a turn has occurred (**720**). If not, the chat is saved (**724**). If so, a determination is made whether a topic shift has occurred (**722**). If not, the chat is saved (**724**). If a topic shift has occurred, IMT **700** goes to step **706**. After the chat is saved, IMT **700** stops (**730**).

[0047] FIG. 8 displays IM Repository system **800** having a search capability, an auto scan capability, and an auto alert capability. Chat transcripts **822** are deposited in repository **820**. Chat transcripts are converted into XML files **834** so that they may be searched using various filters **836** such topic tags or identifiers such as user, date, name and so forth. Alternatively, chat transcripts **822** may be displayed and searched using a browser such as browser **832**. Computer **832** may employ an automatic save program to save chat transcripts to the IM repository. Computer **832** may also employ an auto alert program to identify chat transcripts according to pre-selected criteria and send emails **810** to notify various addresses when the auto alert program identifies matches between topics or identifiers in the repository and topics or identifiers in an auto alert topic table. At the same time, or alternatively, the auto alert program can export the transcript or transcript segment to another format such as e-mail.

[0048] FIG. 9 displays the logic of IM filter (IMF) program **260**. IMF program **260** starts (**902**) and a determination is made whether a full text search is to be made (**910**). If so, a search is conducted (**916**). If not, a determination is made as to whether a filtered search is to be conducted (**912**). If not, IMF **270** returns to step **910**. If so, a filter is chosen (**914**). The search is

conducted (**916**). The search results are viewed (**918**). A determination is made whether the search was satisfactory (**920**). If not, the user enters feedback (**922**). If so, a determination is made whether the user wants to view the full text for the topic found (**924**). If so, the full text for the segment corresponding to the topic found is displayed (**926**). If not, or after viewing the transcript for the search result, a determination is made as to whether another search is to be conducted (**928**). If so, IMF **270** goes to step **910**. If not, IMF **270** stops (**930**). Searches may be client based, web application based, or may be accomplished with VB, Java, Swing, or other such languages.

[0049] Fig. 10 shows the logic of the IM save program (IMS) **280**. IMS **280** begins and a determination is made as to whether all chat transcripts are to be saved (**1010**). If so, IMS goes to step **1018**. If not, a determination is made as to whether some chat transcripts are to be saved (**1012**). If so, a determination is made whether client choice will be allowed (**1014**). If not, the criteria that will be used to select transcripts to be saved is selected (**1016**). If client choice is to be allowed, then IMS **280** goes to step **1018** and transcripts are scanned. Data is received according to the selection criteria (all or some) from the scanned transcripts (**1020**). The received data is converted into XML format (**1022**). A determination is made whether IMS is to continue (**1024**). If so, IMS **280** goes to step **1020**. If not, IMS **280** stops (**1030**).

[0050] FIG. 11 shows the logic of auto alert program IMAL **230** that may be run in conjunction with IMS **280**. IMAL **230**, starts (**1102**) and scans and receives topic tags from scanned transcripts (**1110**). The topic tag is compared to an auto alert table (**1120**). A determination is made as to whether a match has been made (**1130**). If not, IMAL goes to step **1110**. If so, IMAL determines the action to be taken that is associated with the topic in the auto

alert table (**1140**). IMAL executes the action (**1150**). A determination is made whether there is another topic tag (**1160**). If so, IMAL **230** goes to step **1110**. If not, IMAL **230** stops (**1170**).

[0051] With respect to the above description, it is to be realized that the optimum dimensional relationships for the parts of the invention, to include variations in size, materials, shape, form, function, manner of operation, assembly, and use are deemed readily apparent and obvious to one of ordinary skill in the art. The present invention encompasses all equivalent relationships to those illustrated in the drawings and described in the specification. The novel spirit of the present invention is still embodied by reordering or deleting some of the steps contained in this disclosure. The spirit of the invention is not meant to be limited in any way except by proper construction of the following claims.